SPECIFICATION

Docket No. **0544MH-35347**

TO ALL WHOM IT MAY CONCERN:

BE IT KNOWN that WE, Shekhar Iyer and Prashant Soral, citizens of India, residing in the State of Texas, have invented new and useful improvements in a

PRODUCT SUBSTITUTION SEARCH METHOD

of which the following is a specification:

CERTIFICATE OF EXPRESS MAIL
I do hereby certify that the foregoing documents are being deposited with the United States Postal Service as Express Mail, "Post Office to Addressee" in an envelope addressed to the Assistant Commissions for Patents, Washington, D.C. 20231, on this date of June 15, 2000. Name: _____ Express Mail No. : EL643425105US Date of Deposit: June 15, 2000

SYSTEM AND PROCESS FOR PROVIDING PRODUCT ALTERNATES TO PARTNERS IN A MULTI-ENTERPRISE COLLABORATION

TECHNICAL FIELD OF THE INVENTION

This invention relates in general to the field of supply chain, enterprise and site planning and, more particularly, to a system and process for providing product alternates to partners in a multi-enterprise collaboration.

BACKGROUND OF THE INVENTION

Supply chain, enterprise and site planning applications and environments are widely used by manufacturing industries for decision support and to help manage complex manufacturing operations. In contrast with conventional planning software applications that provide a monolithic application architecture, products are becoming available to support multi-domain, distributed and heterogenous planning environments that characterize present day supply chains. One such product is the RHYTHM COLLABORATION suite of software products available from i2 Technologies, Inc.

An inherent characteristic of such products, and in particular the multi-domain heterogenous architecture of the RHYTHM COLLABORATION suite, is that they support products spanning multiple domains. The products supplied by the seller domain in the collaboration are consumed by the buyer domain. It is

not uncommon that the product supply is constrained and does not meet the buyer demand. This, in general, leads to lost sales and customer dissatisfaction for the seller domain and can be further detrimental to the multi-enterprise supply chain as the buyer domain may not be able to satisfy its own customers in a timely fashion.

A solution to the problem is to provide the buyer domain with a list of viable alternate products that can be consumed instead of the original product. This will also extend the domain of the decision support across multiple enterprises in the supply chain. It is desirable for decision support software to cover even larger domains in the decision making process because typically, the larger the domain of the decision support, the more optimal the decision will be. It is the purpose of the present invention to develop a system and process for providing such product alternates to partners in a multi-enterprise collaborative supply chain.

SUMMARY OF THE INVENTION

In accordance with the present invention, a system and process for providing product alternates to partners in a multi-enterprise collaborative environment are disclosed that result in increased customer satisfaction, improved demand fulfillment and optimal decision making across multiple enterprises.

A computer implemented module within the RHYTHM COLLABORATION suite of products is provided for identifying the best products among the alternates available. The process involves a two-step approach. In the first step, a graphical user interface is used to gather product requirements from the user. This includes specification of the level of similarity between the original product and the alternate products required by the user. Based on the user's input of the product requirements and the desired similarity level, the process employs an algorithm to retrieve alternate products that satisfy the user requirements when compared to the original product. The process involves ranking the alternate products in order of their similarity to the original product. For each of the product characteristics identified by the user in the first search step, the algorithm is used to conduct a detailed similarity analysis which compares the more granular attributes because of which a product possesses the particular characteristic. The result of executing the sort algorithm is the generation of a global index value for each alternate product with respect to the original product. The alternate products are then ranked based on the global index. Finally, a graphical user interface presents the list of alternate products to the user.

In practice, there generally exist numerous feasible alternates to a product and in the absence of an expert system, such as that detailed in the present invention, the user must rely on subjective judgment and experience for trimming the set of feasible alternates to a manageable number. There is a high

probability that the optimum alternates will be eliminated in such a subjective process. The system and process in the present invention provide a systematic procedure that will objectively select the best feasible alternates for a product. Another technical advantage of the present invention is that it eliminates the need for the user to know the details of the methodology and performs the search and sort operations based on user specifications through a menu operated graphical user interface.

Additional technical advantages should be readily apparent to one skilled in the art from the following figures, descriptions, and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete understanding of the present invention and advantages thereof may be acquired by referring to the following description taken in conjunction with the accompanying drawings, in which like reference numbers indicate like features, and wherein:

FIGURE 1 is a diagram showing an overview of the process used to identify alternate products; and

FIGURE 2 is a diagram that demonstrates calculation of the Attribute Similarity Index.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

In important objective of the present invention is to identify alternate products with adequate supply, which are similar to a candidate product with inadequate supply. A two-step procedure is adopted in the system in order to make it efficient and practical. The first step rapidly identifies a set of potentially useful alternate products based on product attributes that are desired in the candidate product. The second step evaluates only those alternate products that have been identified in the first step and ranks them based on an analysis of detailed product attributes. Thus the user can review only the few most relevant alternates and satisfy demand for the candidate by increasing the forecasted quantity for one or more of the alternates.

Referring to Figure 1, a user enters requests at some type of user interface 10 as known in the art. The details of the interface 10 are not important to the present invention, and many different known interfaces are suitable for use. A search engine 12 operates to identify similar products, in turn utilizing a database 14 of detailed product information. Operation of the search engine 12 is described in detail below. Search engine 12 generates an output 16 that contains a list of similar parts according to the criteria set forth below. This is not yet presented to the user.

A sort engine 18 accepts as input the output 16 from search engine 12, and sorts the list of similar parts in an order that is most likely acceptable to the user. Details of this process are also described below. Once the preferred order is determined, the sorted list is presented to the user through user interface 20. In many cases, interfaces 10 and 20 will actually be one interface, even though they are separated in Figure 1 for conceptual clarity.

The user is presented with a broad range of product characteristics. These characteristics are specific to the manufacturing environment being dealt with. The methodology is demonstrated here by applying it to the computer industry. Broad product characteristics for computers may include processing speed, multi-media support, data storage etc. Through a graphical interface, the user inputs the search intent by specifying the desired characteristics that the search is to be based upon, and the level of similarity for each product characteristic.

Each product characteristic is broken down into elementary components known as attributes. For example, the processing speed can be broken down into the internal clock speed for the CPU, the bus speed, the type and size of RAM. Consider products P_i and P_j. X denotes the set of all attributes for a product e.g. a computer may have the following set of attributes: X = {Processor Speed, Memory Size, Hard File Size, Monitor, CDROM, Operating System}. Two types of attributes can be identified:

a. Continuous valued attributes: The values of this class of attributes bear a decisive resemblance with each other, and the magnitude of the attribute similarity index signifies the degree of resemblance. Examples may include processor speed, RAM size, and hard disk size.

b. Binary valued attributes: The values of this class of attributes are unique and no consistent similarity can be identified among them. Examples may include distinct types of operating systems.

X_m is an element of the set X. $V(X_m P_j)$ is the value of attribute m for product j. In trying to assess whether a product can act as a substitute for another product, the value of each attribute of the candidate substitute is compared with the value of the corresponding attribute for the original product.

The measure used to quantify the similarity of a substitute to the original product in terms of a specific attribute is called the Attribute Similarity Index. Assume that a user is trying to determine if product P_j is a substitute for P_i. In addition, the products are being compared with respect to attribute X_m. The Attribute Similarity Index is denoted as $SI_{ij}(X_m)$. The first index (i) stands for the original product P_i and the second one (j) for the substitute product P_j. The fact that this similarity index is for a specific attribute is denoted by X_m in brackets.

The measure used to quantify the similarity of a substitute to the original product taking all attributes into consideration is called the Product Similarity Index. Assume that a user is trying to determine if P_j is a substitute for P_i. The Product Similarity Index is denoted as $\mathbf{SI_{ij}}$. The first index (i) stands for the original product P_i and the second one (j) for the substitute product P_j. The Product Similarity Index is a weighted sum of Attribute Similarity Indices. Each attribute is assigned a weight. The weights are input obtained from the user through a graphical interface.

Algorithm

1. For each continuous valued attribute in the set X, the maximum and minimum values over all products are determined.

2. Consider a single product, P_i. Calculate Product Similarity Indices, S_{ij}, for all other products, $P_{j,j\neq i}$. Steps involved in doing so are given below.
    a. Consider a product P_j other than P_i.
    b. The Attribute Similarity Index $\mathbf{SI_{ij}(X_m)}$ is calculated for each attribute in the set X.
    c. The Product Similarity Index $\mathbf{SI_{ij}}$ is calculated.
    d. $\mathbf{SI_{ij}}$ is compared with a user-defined threshold value for Product Similarity Index $\mathbf{SI_{i,\,min}}$. $\mathbf{SI_{i,\,min}}$ holds for any product relative to original product P_i.

e. Include P_j as a substitute only if the Product Similarity Index exceeds $SI_{i,\,min}$.

3. The list generated in step 2 is arranged in descending order with respect to Product Similarity Index. In case of a tie with respect to Product Similarity Index, the products will be ranked with respect to the Attribute Similarity Index for the attribute with the highest weight. If even this does not eliminate the tie, the attribute with the second highest weight will be used and so on. The final ordered list of substitutes for product P_i will be denoted as $\mathbb{P}_{i,\,Subs}$.

4. Generate the list for all products.

Similarity Index Calculation

Attribute Similarity Index for Attributes with Numerical Values

Consider the following scenario. The user requests substitutes for product P_O. Assume product P_O has an attribute X_A with value $V(X_A P_O) = A_O$. The maximum and minimum attribute values for attribute X_A have already been generated and are equal to $V_{max}(X_A) = A_{max}$ and $V_{min}(X_A) = A_{min}$ respectively. Product P_Q and Product P_R are being considered as substitutes for Product P_O. The value of attribute X_A for Product P_Q is $V(X_A P_Q) = A_Q$ and A_R for Product P_R.

The system will compare the attribute value for the substitute, A_Q with that for the original product, A_O. If A_Q is less than A_O, the Attribute Similarity Index for Product P_Q and attribute X_A, $\mathbf{SI_{OQ}(X_A)}$ is calculated as follows:

$$SI_{OQ}(X_A) = \frac{x^2 - A^2_{\min}}{A_o^2 - A^2_{\min}}$$

If A_Q is greater than A_O, the Attribute Similarity Index for Product P_Q and attribute X_A, $\mathbf{SI_{OQ}(X_A)}$ is calculated as follows:

$$SI_{OQ}(X_A) = \frac{x^2 - A^2_{\max}}{A_o^2 - A^2_{\max}}$$

If A_Q equals A_O, the Attribute Similarity Index for Product P_Q and attribute X_A, $\mathbf{SI_{OQ}(X_A)}$ equals 1.

As an example, consider FIGURE 2, which is a graph of the Attribute Similarity Index for the "Processor" attribute. Here A_O is 233 MH_Z for which the Product Attribute Similarity Index is 1. A_{max} is 400 MH_Z and A_{min} is 133 MH_Z. The following equations hold:

$$SI_{OQ}(X_{Processor}) = \frac{x^2 - 133^2}{233^2 - 133^2} \qquad \textit{for } 133 \le x < 233$$

$$SI_{OQ}(X_{Processor}) = 1 \qquad \text{for } x = 233$$

$$SI_{OQ}(X_{Processor}) = \frac{x^2 - 400^2}{233^2 - 400^2} \qquad \text{for } 233 < x \leq 400$$

Attribute Similarity Index for Binary Valued Attributes

If the attribute value for a substitute is not the same as that for the original product, the Attribute Similarity Index for the substitute is 0. It is 1 if the attribute value for the substitute is equal to that for the original.

Product Similarity Index

The Product Similarity Index for Product P_Q is calculated as follows:

$$SI_{OQ} = SI_{OQ}(X_{Processor}) \times W_{Processor}\} + \{SI_{OQ}(X_{Memory}) \times W_{Memory}\} + \{SI_{OQ}(X_{HD}) \times W_{HD}\} + \{SI_{OQ}(X_{CDROM}) \times W_{CDROM}\} + \{SI_{OQ}(X_{Monitor}) \times W_{Monitor}\} + \{SI_{OQ}(X_{OS}) \times W_{OS}\}$$

The above assumes that the weights are normalized to one, i.e. $\sum_i W_i = 1$.

The list of substitute products for each original product is shortened to include only those substitute products for which excess supply is available. The supply information can be obtained from an available advanced planning engine such as i2 TECHNOLOGIES' Supply Chain Planner application. The user is presented with the desired number of top-ranking substitutes in the list.

Extension of the Methodology

An important aspect of the present methodology is extensibility. Areas in which the present invention can be extended include the use of a more extensive analytic process to arrive at the weights for the selection of alternate products, the presentation of broad product characteristics instead of detailed product attributes to the user, and the elimination of the need to calculate substitutes for all products in the first stage of the method described above.

Although the present invention has been described in detail, it should be understood that various changes, substitutions can be made hereto without departing from the spirit and scope of the invention as defined by the appended claims.